

Sieci neuronowe - wprowadzenie

Czym jest sztuczny neuron?

- Sztuczny neuron to uproszczony model neuronu biologicznego.
- Przetwarza dane wejściowe, mnoży przez wagi i dodaje odchylenie (ang. bias).
- Wynik przechodzi przez funkcje aktywacji, tworząc wyjście.

Wzór działania neuronu

Wyjście = $f(w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n + b)$

$x_1, x_2 \dots$ - dane wejściowe, $w_1, w_2 \dots$ - wagi, b - bias

f - funkcja aktywacji, np. sigmoid, tanh, ReLU

Budowa sieci neuronowej

Warstwa wejściowa: dane

Warstwy ukryte: przetwarzanie

Warstwa wyjściowa: wynik

Funkcje aktywacji

- Sigmoid - zwraca wartości od 0 do 1
- ReLU - tylko dodatnie wartości
- tanh - wartości od -1 do 1

Uczenie sieci (backpropagation) to podstawowy mechanizm, dzięki któremu sieć neuronowa "uczy się" na błędach i stopniowo poprawia swoje działanie:

Sieć uczy się na podstawie danych i błędu.

Błędy są „cofane” i korygują wagi.

Powtarzamy aż model działa lepiej.

Przykładowo klasyfikacja dla zbioru 100 recenzji tekstowych usługi linii PLL LOT przy użyciu sztucznej sieci neuronowej:

1. Sieć dostaje dane wejściowe (recenzję klienta)

Na podstawie tych danych dokonuje prognozy: czy klient poleci firmę ("tak" lub "nie").

Założmy, że sieć dostaje dane wejściowe i na ich podstawie zwraca wynik:

$\hat{y} = 0.8$ (czyli przewiduje z 80% pewnością, że odpowiedź to "tak").

2. Porównujemy wynik sieci z wartością rzeczywistą

Sprawdzamy, czy sieć się pomyliła i obliczamy błąd (loss).

Zakładamy, że w rzeczywistości poprawna odpowiedź to:

$y = 1$ (czyli rzeczywiście „tak”).

Obliczamy błąd np. średniokwadratowy (MSE)

MSE to miara błędu średniokwadratowego, czyli średniej z podniesionych do kwadratu różnic między wartościami rzeczywistymi (y) a przewidywanymi (\hat{y}).

$$\text{Błąd} = (y - \hat{y})^2 = (1 - 0.8)^2 = 0.04$$

Ten błąd mówi, jak bardzo sieć się myli. Celem uczenia jest **zmniejszanie błędu w kolejnych krokach**.

Wartość 0.04 oznacza niewielki błąd (80% pewności zamiast 100%). Im mniejsza wartość MSE, tym lepiej. Wartość bliska 0 oznacza, że przewidywania modelu są bardzo dokładne.

3. Cofanie błędu (backpropagation)

Błąd jest „przesyłany wstecz” przez wszystkie warstwy sieci. Dzięki temu każda **waga** (czyli połączenie między neuronami) „dowiaduje się”, jak bardzo przyczyniła się do błędu.

Backpropagation wykorzystuje błąd do:

- **obliczenia pochodnych** (czyli jak silny był wpływ każdej wagi na błąd)
- **zmiany wag** w odpowiednim kierunku (np. za pomocą metody gradient descent)

4. Następuje korekta wag (uczenie sieci)

Wagi są odpowiednio modyfikowane (jeśli coś było za silne lub za słabe, zostaje to skorygowane).

Każda waga jest aktualizowana według wzoru:

$$w_{\text{nowe}} = w_{\text{stare}} - \eta \cdot \frac{\partial \text{Loss}}{\partial w}$$

gdzie:

- η to **learning rate** – mały krok uczenia (np. 0.01),
- $\frac{\partial \text{Loss}}{\partial w}$ to pochodna błędu względem wagi.

5. Proces się powtarza

Proces powtarzamy dla wszystkich kolejnych 100 przykładów.

Z czasem błąd będzie bardzo mały, gdyż sieć uczy się, jak najlepiej odpowiadać i popełnia coraz mniej błędów.